Documentação do Projeto - Nutri Ai

# 🔹 Nome do Projeto:

Nutri Ai  
  
Aplicativo de apoio nutricional, onde o usuário pode registrar sua dieta atual (caso tenha acompanhamento profissional) ou receber orientações básicas de nutrição com base em perguntas guiadas. O app também oferece um chat com inteligência artificial (GPT API) para sugerir trocas de refeições e receitas fitness personalizadas.

# 📁 Estrutura de Pastas

NutriAi/  
│  
├── App.tsx # Ponto de entrada da aplicação  
├── package.json # Informações e dependências do projeto  
├── tsconfig.json # Configuração do TypeScript  
├── .gitignore # Arquivos ignorados no controle de versão  
│  
└── src/ # Código-fonte principal do app  
 ├── screens/ # Telas principais da aplicação  
 │ └── HomeScreen.tsx # Tela inicial (Home)  
 ├── components/ # Componentes reutilizáveis  
 ├── services/ # Requisições HTTP, APIs, etc.  
 ├── context/ # Estados globais usando Context API  
 ├── utils/ # Funções auxiliares  
 ├── data/ # Dados mockados ou estáticos  
 ├── styles/ # Estilos globais e temas  
 └── navigation/ # Configuração de navegação

# 🧠 Tecnologias utilizadas

|  |  |
| --- | --- |
| Tecnologia | Função |
| React Native | Framework para desenvolvimento mobile multiplataforma |
| Expo | Ambiente simplificado para desenvolvimento React Native |
| TypeScript | Superset de JavaScript com tipagem estática |
| VS Code | Editor de código |
| Node.js | Runtime JS utilizado pelo Expo |
| Git/GitHub | Versionamento de código |

# 🔧 Funcionalidade atual

- Tela inicial (HomeScreen.tsx) implementada, com visualização básica.  
- App configurado para exibir a HomeScreen diretamente via App.tsx.  
- Estrutura de projeto modular criada, com separação por responsabilidade.  
- Git inicializado e conectado ao repositório remoto do GitHub:  
 - Usuário GitHub: lucca7r  
 - Repositório remoto: https://github.com/lucca7r/NutriAi.git

# ⚙️ Ambiente de Execução

O projeto é executado com Expo Go no celular ou emulador.  
  
Inicialização do app via terminal com:  
**npx expo start**

# 📝 Observações

- A navegação entre telas ainda será implementada (React Navigation).  
- A integração com a API do GPT (para chat nutricional) será feita na fase posterior.  
- Todo o projeto está em TypeScript para garantir tipagem e escalabilidade.

## 📦 Dependências necessárias

Antes de rodar o projeto, certifique-se de instalar todas as dependências executando este comando no terminal, dentro da pasta do projeto:

npm install

Ou, se quiser garantir todos os pacotes desde o início, pode executar diretamente:

npm install \  
 react react-native \  
 expo \  
 typescript \  
 @types/react @types/react-native \  
 @react-navigation/native \  
 @react-navigation/native-stack \  
 react-native-screens \  
 react-native-safe-area-context \  
 react-native-gesture-handler \  
 react-native-reanimated \  
 react-native-vector-icons \  
 expo-font \  
 expo-status-bar \  
 expo-splash-screen \  
 expo-linear-gradient

Para o TypeScript funcionar corretamente com o React Navigation:

npm install -D @types/react @types/react-native

Além disso, **(caso nao esteja atualizado confira)** para evitar os erros de tipo (.d.ts) no VS Code, atualize seu tsconfig.json com:

{  
 "compilerOptions": {  
 "moduleResolution": "node",  
 "target": "ESNext",  
 "module": "ESNext",  
 "jsx": "react-native",  
 "strict": true,  
 "esModuleInterop": true,  
 "skipLibCheck": true,  
 "allowSyntheticDefaultImports": true,  
 "resolveJsonModule": true,  
 "forceConsistentCasingInFileNames": true  
 },  
 "include": ["src", "@types"]  
}